Bahasa Pemprogaman Python

Python menjadi salah satu bahasa pemrograman yang terkenal karena kemampuannya dalam mengeksekusi sejumlah instruksi secara langsung, yang bersifat interpretatif. Guido van Rossum memperkenalkan Python pada tahun 1991. Bahasa pemrograman ini mencakup sintaks, struktur data, operasi, dan pemrograman berorientasi objek. Selain itu, terdapat juga library atau modul yang sering digunakan dalam berbagai aplikasi, seperti Pandas dan NumPy (Priyatno dkk., 2023).

Pandas

Pandas adalah library Python open source yang sering digunakan untuk menganalisis data dan membangun machine learning. Library ini dikembangkan oleh Wes McKinney pada 2008 saat ia bekerja di AQR Capital Management dan kemudian diperbolehkan sebagai open source (VanderPlas, 2016). Pandas memiliki banyak fitur yang membantu dalam manipulasi data, seperti memfilter data, mengelompokkan dan memisahkan data menurut preferensi, dan lain sebagainya. Berikut adalah beberapa fungsi yang digunakan untuk pengolahan data di penelitian ini serta pengertiannya menurut VanderPlas (2016).

* pd.read\_excel() : Metode ini digunakan untuk membaca file Excel dan mengimpor dataannya ke dalam objek DataFrame. Fungsi ini mendukung berbagai ekstensi file Excel, seperti .xls, .xlsx, .xlsml, .xltx, dan lainnya.

Contoh penggunaan :

df = pd.read\_excel(“file\_excel.xlsx”)

* Pd.head() : Metode ini digunakan untuk menampilkan N baris pertama dari sebuah DataFrame. Secara default, metode ini akan menampilkan lima baris pertama dari DataFrame. Metode ini menerima satu parameter, yaitu jumlah baris yang ingin ditampilkan.

Contoh penggunaan :

df = pd.read\_excel(“file\_excel.xlsx”)

df.head()

* Pd.iloc[] : Metode ini digunakan untuk mengambil item pada dataframe berdasarkan indeks baris dan kolom. Nilai list pertama menunjukkan baris mana yang akan diambil, dan nilai pada list kedua menunjukkan kolom mana yang akan diambil. Pada bahasa python, indeks dimulai dari angka 0.

Contoh penggunaan :

Baris\_pertama\_kolom pertama = df.iloc[0,0]

* pd.div() : Metode ini digunakan untuk melakukan operasi pembagian pada DataFrame. Operasi ini bersifat berbasis elemen, sama dengan operator pembagian “/”. Dalam kasus ini, setiap nilai dalam DataFrame dibagi dengan nilai yang ditentukan. Nilai yang ditentukan dapat berupa bilangan konstan, objek berstruktur data, Pandas Series, atau DataFrame lain yang sesuai dengan DataFrame asli.

Contoh penggunaan :

Df\_A = df\_A.div(df\_input.iloc[0,:], axis=1)

Numpy

Numpy merupakan salah satu libary bahasa pemrograman Python yang diciptakan pada tahun 2005 oleh Travis Olliphant. Tujuan utama library ini adalah digunakan dalam manipulasi data numerik karena menyediakan struktur data array yang efisien untuk menyimpan dan memanipulasi data multidimensi. Keunggulan libarry ini terdapat pada operasi matematika vektorisasi, efisiensi operasi numerik tinggi, dan kemampuannya dalam komputasi linier serta aljabar (Albon, 2018).

* values : Metode ini digunakan untuk mendapatkan representasi NumPy dari DataFrame yang ada. Metode ini mengembalikan semua nilai dalam DataFrame sebagai array NumPy, yang dapat digunakan untuk berbagai aplikasi seperti analisis data, manipulasi, dan visualisasi

Contoh Penggunaan :

Matriks\_A = df.values

* identity() : Metode identity digunakan untuk membuat matriks identitas dengan ukuran n kali n

Contoh Penggunaan :

Matriks\_identitas = np.identity(17)

* linalg.inv() : Metode linalg.inv digunakan untuk melakukan operasi inverse matriks pada matriks yang telah diinputkan.

Contoh Penggunaan :

Matriks\_I\_A = np.linalg.inv(Matriks\_A)

* Sum() : Metode sum digunakan untuk melakuakn operasi penjumlahan pada tiap baris atau kolom. Dengan menggunakan parameter axis = 0, maka penjumlahan akan dilakukan pada setiap baris, sedangkan axis = 1 dilakukan pada setiap kolom.

Contoh Penggunaan :

Keterkaitan\_kebelakang = Matriks\_I\_A.sum(axis = 0)

Keterkaitan\_kedepan = Matriks\_I\_A.sum(axis = 1)

* Mean() : Sama seperti metode sum, metode mean digunakan untuk melakukan operasi rata-rata pada suatu matriks pada tiap baris atau kolom.

Rataan\_kebelakang = Keterkaitan\_kebelakang.mean(axis = 1)

Rataan\_kedepan = Keterkaitan\_kedepan.mean(axis = 0)

* Reshape() : Metode reshape digunakan untuk mengubah bentuk matriks atau dimensi matriks tanpa mengubah nilai dari matriks tersebut.

Contoh Penggunaan :

matrix\_I\_A\_inv.sum(axis=1).reshape(17,1)

* Operasi matriks menggunakan numpy dapat dilakukan dengan menggunakan operator “+”, “-“.

Contoh Penggunaan :

Matriks\_A\_plus\_B = Matriks\_A + Matriks\_B

Matriks\_A\_min\_B = Matriks\_A – Matriks\_B

* Dot() : Metode dot digunakan untuk melakukan operasi perkalian antar matriks.

Contoh Penggunaan :

np.dot(matriks\_A,matrix\_B)

* Multiply() : Metode multiply digunakan untuk melakukan perkalian matriks terhadap konstanta.

Contoh Penggunaan :

Matriks\_A.multiply(100)

Matriks\_B.multuply(1000)

Referensi

Priyatno, A. M., Firmanda, F. I., Farhas, R. J., Amalia, F., & Sudirman, W. F. R. (2023). Pelatihan Data Science menggunakan Bahasa Pemrograman Python di PT Ilmu Data Indonesia. *Jurnal Pengabdian Pendidikan Dan Teknologi Masyarakat FKIP*, *1*(1), 31–36. <https://doi.org/10.31004/dedikasi.v1i1.12>

VanderPlas, J. (2016). *Python Data Science Handbook: Essential Tools for Working with Data*. “O’Reilly Media, Inc.”

Albon, C. (2018). *Machine Learning with Python Cookbook: Practical Solutions from Preprocessing to Deep Learning*. “O’Reilly Media, Inc.”